MOTSynth: How Can Synthetic Data Help Pedestrian Detection and Tracking?

Matteo Fabbri[1,3] Guillem Brasó[2] Gianluca Maugeri[1] Orcun Cetintas[2] Riccardo Gasparini[1,3]
Aljoša Ošep[2] Simone Calderara[1] Laura Leal-Taixé[2] Rita Cucchiara[1]
[1]University of Modena and Reggio Emilia, Italy [2]Technical University of Munich, Germany
[1]{firstname.lastname}@unimore.it [2]{firstname.lastname}@tum.de
[3]GoatAI S.r.l.
[3]{firstname.lastname}@goatai.it

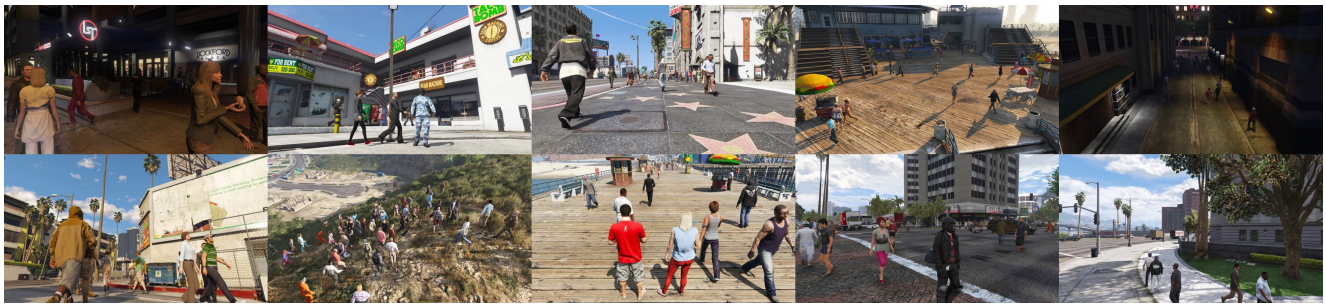


Figure 1: We propose *MOTSynth*, a large and diverse dataset for pedestrian detection, re-identification and multi-object tracking. Due to high diversity, we are able to obtain state-of-the art performance by training models solely on synthetic data.

Abstract

Deep learning-based methods for video pedestrian detection and tracking require large volumes of training data to achieve good performance. However, data acquisition in crowded public environments raises data privacy concerns – we are not allowed to simply record and store data without the explicit consent of all participants. Furthermore, the annotation of such data for computer vision applications usually requires a substantial amount of manual effort, especially in the video domain. Labeling instances of pedestrians in highly crowded scenarios can be challenging even for human annotators and may introduce errors in the training data. In this paper, we study how we can advance different aspects of multi-person tracking using solely synthetic data. To this end, we generate MOTSynth, a large, highly diverse synthetic dataset for object detection and tracking using a rendering game engine. Our experiments show that MOTSynth can be used as a replacement for real data on tasks such as pedestrian detection, re-identification, segmentation, and tracking.

1. Introduction

Object detection and tracking in crowded real-world scenarios are challenging and difficult problems with long-standing research history, with applications ranging from autonomous driving to visual surveillance. Since the advent of deep learning, the community has been investigating how to effectively leverage neural networks [41, 45, 54, 69, 13, 65, 42, 25, 75, 6, 10, 78, 80, 38] to advance the field. However, all these approaches are data-hungry, and data collection and labeling are notoriously difficult and expensive. Moreover, dataset collection in public environments[1] raises privacy concerns. In fact, European Union already passed privacy-protecting laws such as General Data Protection Regulations (GDPR [2]) to protect the privacy of its citizens that prohibit the acquisition of personal visual data without authorization; ethical issues regarding privacy are also critical in the US, where datasets for training person re-identification modules such as DukeMTMC [62] were taken offline due to privacy concerns [33].

A possible solution for the aforementioned issues is to employ virtual worlds. The community has already recognized the potential of synthetic data, successfully used for benchmarking [44] or to compensate for the lack of training data [3, 9]. To the best of our knowledge, so far, synthetic data could fully replace recorded data only for low-level tasks such as optical flow estimation [21]. For higher-level tasks, such as object detection, tracking and segmentation,

[1]Crowded public scenes are especially difficult to record during the COVID-19 pandemic.

existing methods usually need mixed synthetic and real data and employ alternate training scheme [3] or domain adaptation [9] and randomization [72] techniques.

In this paper, we aim to answer a challenging question: *Can we advance state-of-the-art methods in pedestrian detection and tracking using only synthetic data?* To this end, we created *MOTSynth*, a large synthetic dataset for pedestrian detection, tracking, and segmentation, designed to replace recorded data. *MOTSynth* comes in a bundle with temporally consistent bounding boxes and instance segmentation labels, pose occlusion information, and depth maps. As shown in the field of robot reinforcement learning [72] and vision [73], synthetic datasets should significantly vary in terms of lighting, pose, and textures to ensure that the neural network learns all invariances present in the real world. Based on these insights, we generate a large and diverse dataset that varies in terms of environments, camera viewpoints, object textures, lighting conditions, weather, seasonal changes, and object identities (see Fig. 1). Our experimental evaluation confirms that diversity plays a pivotal role in bridging the synthetic-to-real gap.

The main focus of our study is on *how MOTSynth* can help us to advance pedestrian detection, re-identification, and tracking by studying how different aspects of these tasks can benefit from our data. To this end, we first train several state-of-the-art models for *pedestrian detection*, *segmentation*, *re-identification*, *frame-to-frame regression* and *association* on synthetic data and evaluate their performance on the real-world pedestrian tracking dataset MOTChallenge [18]. Our experiments show that models, trained on synthetic data are on-par with state-of-the-art on MOTChallenge MOT17&MOT20, while extremely crowded MOT20 still require fine-tuning. Second, we show that prior synthetic datasets [24, 43] are not suitable for bridging the synth-to-real gap for the task of pedestrian detection and tracking. Moreover, we confirm that the diversity in *MOTSynth* is a key for bridging this gap – and is far more important than the sheer amount of data. In addition to a thorough experimental analysis, *MOTSynth* also opens the door to future research on how different components, such as depth and human pose, can be used to advance multi-object tracking in a well-controlled environment.

To summarize, the main contributions of this paper are the following: (i) we open source the largest synthetic dataset for pedestrian detection and tracking with more than 1.3 million densely annotated frames and 40 million pedestrian instances; (ii) we show that such a diverse dataset can be a complete substitute for real-world data for high-level tasks such as pedestrian detection and tracking in several scenarios, as well as re-identification and tracking with segmentation; (iii) we provide a comprehensive analysis on how such synthetic worlds can be used to advance the state-of-the-art in pedestrian tracking and detection.

2. Related Work

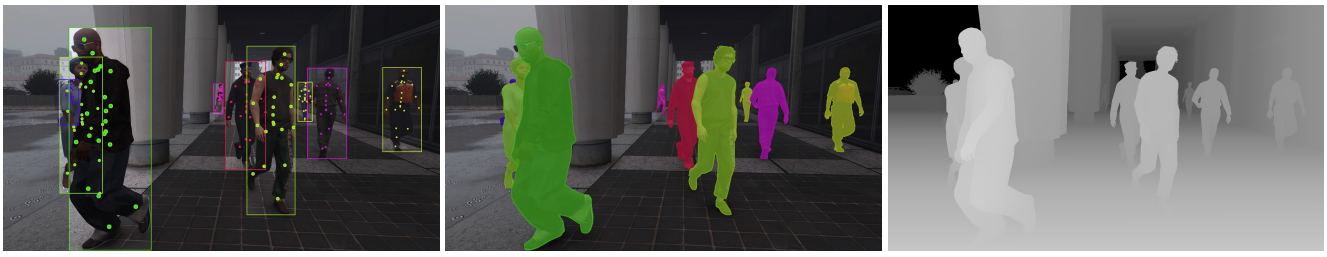
Advances in computer vision have been driven by the constant growth of available datasets and benchmarks, such as Pascal VOC [22], ImageNet [64], COCO [49], CityScapes [15] and MOTChallenge [18].

Multi-object tracking (MOT). In terms of autonomous driving, the pioneering MOT benchmark is the KITTI benchmark [28] that provides labels for object detection and tracking in the form of bounding boxes and segmentation masks [75]. However, sequences were collected in a single city in clear weather conditions from a camera mounted on a car. The recently proposed BDD100k [81] covers over 100K videos with high geographic, environmental, and weather diversity. Several recent automotive tracking datasets and benchmarks are LiDAR-centric, providing labels in form of 3D bounding boxes [12, 57, 70]. The recently proposed TAO dataset [17] provides bounding box labels for over 800 object classes.

Visual surveillance centric datasets focus on crowded scenarios where pedestrians are interacting and often occluding each other. MOTChallenge [18] benchmark suite played a pivotal role in benchmarking multi-object tracking methods and providing consistently labeled crowded tracking sequences. In particular, MOT17 [54] provides challenging sequences of crowded urban scenes, capturing severe occlusions and scale variations. MOTS [75] The latest release, MOT20 [19] pushes the limits by providing labeled sequences captured in extremely dense scenarios. In terms of car surveillance, UA-DETRAC [77] consists of 100 sequences recorded from a high viewpoint with the goal of vehicle tracking.

Object tracking is deeply entwined with person re-identification (ReID), as several state-of-the-art tracking methods [6, 10] rely on learned ReID features. Since DukeMTMC dataset was taken offline due to privacy concerns [33], the most commonly used ReID datasets are Market1501 [83] and CUHK03 [47]. With this work, we aim to replace recorded data for training object detection, re-identification, and tracking with synthetic data.

Synthetic datasets. Data collection usually demands a tremendous amount of manual work. As more data is constantly required to train ever-growing models, the cost of labeling such datasets is becoming prohibitive. This burden can either limit the quality or the quantity of available data and hinder progress. A possible solution to the aforementioned problems is to employ virtual worlds. Such simulated environments have been successfully applied to low-level tasks, such as feature descriptor computation [40], visual odometry [30, 32, 82, 60], optical flow estimation [5, 11, 60, 52, 44, 53] and depth estimation [44, 53]. Simulated worlds have also been recently utilized for higher-level tasks like semantic segmen-



(a) Bounding boxes and pose (b) Segmentation masks (c) Depth maps

Figure 2: *MOTSynth* labels. From left to right: bounding boxes and pose, instance segmentation masks, and depth. Best viewed on screen.

tation [74, 31, 63, 37, 60, 44, 61, 43], multi-object tracking [27, 24, 71, 36], hand tracking [67], human pose estimation [68, 24, 29, 23], pedestrian and car detection [51, 3, 39], and as virtual environments for robot reinforcement learning [72]. The aforementioned works mainly leverage synthetic data for evaluation in scenarios where precise ground-truth data is difficult to obtain [44] or as means for pre-training data-hungry deep learning models. However, apart from optical flow [21], none of those attempts of using simulated environments was able to replace manually labeled data completely. In contrast, we focus on bridging the synth-to-real gap for pedestrian detection, ReID, and tracking and perform a thorough analysis of the effect of the amount of training data vs. diversity.

3. *MOTSynth* Dataset

MOTSynth is a large, synthetic dataset specifically designed for training models for pedestrian detection, tracking and segmentation. In the following, we detail the dataset generation process (Sec. 3.1) and perform statistical analysis and comparison to other real-world and synthetic datasets (Sec. 3.2).

3.1. Dataset Generation

To generate *MOTSynth*, we follow prior work [60, 43, 24] and we utilize Grand Theft Auto V (GTA-V) videogame, which simulates a city and its inhabitants in a three-dimensional world. More precisely, we utilized Script Hook V library [1]

Setting up screenplays. The first part of recording generation is the scenario (scene) creation. To this end, we manually explored $130km^2$ (about an eighth of Los Angeles County) of the GTA-V virtual world. To generate screenplays, we manually placed camera viewpoints to selected scenarios and set people behavior-related settings, such as the number of pedestrians per scenario, performed actions (such as *standing*, *sitting* or *running*), and paths traveled. In order to simulate dynamics specific to the most crowded areas, we manually pre-planned pedestrian flows by defining a set of trajectories that groups of pedestrians have to follow. We relied on the collision avoidance algorithm to obtain natural pedestrian behavior for each agent. For this step, we utilized the mod proposed in [24] in order to optimize the process. The screenplay generation was the only manual procedure in the creation of *MOTSynth* and took in total only 16 hours. To obtain diverse actors, we randomly varied generative attributes of 579 pedestrian models, provided by the GTA-V game, *e.g.*, different *clothes*, *backpacks*, *bags*, *masks*, *hair* and *beard styles*, yielding over $9,519$ unique pedestrian identities in total. Thus, our generated pedestrians are suitable for training ReID models. We manually set 256 screenplays and combined them with 128 screenplays from [24][2], totalling 384 screenplays.

Rendering. After setting up screenplays, we can simulate virtual world dynamics and render different views of the simulated environments. To obtain as diverse renderings as possible, we randomized weather conditions and daytime of the recordings. Weather conditions captured on our dataset are *clear, extra sunny, cloudy, overcast, rainy, thunder, smog, foggy, and blizzard*. We recorded each screenplay twice, one during the day and one during the night, totaling 768 generated diverse sequences.

Label generation. Every clip comes with a precise 3D annotation of visible and occluded body parts, temporally consistent 2D bounding boxes and segmentation mask labels for pedestrians, and depth maps (see Fig. 2). While we do not exploit depth maps in this work, these are cues often used in MOT [46, 56, 36, 50]. Hence, we believe they can be used to further advance the field. In terms of completeness, *MOTSynth* exceeds any other dataset in terms of scenario variability, number of entities, and types of annotations.

3.2. Statistical Analysis

MOTSynth sequences were rendered as a Full HD video at 25 FPS. Each video sequence contains 29.5 people per frame on average, with a maximum of 125 people, totaling more than $40M$ bounding boxes over $1.3M$ densely annotated frames. The distance of the actors from the camera

[2]We thank the authors of [24] for sharing their screenplays.

Dataset	#Frames	#Inst.	3D	Pose	Segm.	Depth
PoseTrack [4]	46k	276k		✓		
MOTS [75]	2k	26k			✓	
MOT-17 [54]	11k	292k				
MOT-20 [19]	13k	1,652k				
VIPER [43]	254k	2,750k	✓		✓	
GTA [44]	250k	3,875k			✓	✓
JTA [24]	460k	15,341k	✓	✓		
MOTSynth	1,382k	40,780k	✓	✓	✓	✓

Table 1: Overview of the publicly available datasets for pedestrian detection and tracking. For each dataset, we report the numbers of annotated frames and instances, as well as the availability of different labels.

ranges between 0 and 101 meters, resulting in (projected) bounding boxes heights between 0 and $1,080$ pixels.

We split *MOTSynth* into training and validation sets, containing 576 and 192 clips, respectively. We ensured that these splits were roughly balanced in terms of weather conditions, daytime, and density and that no unique person identity appears across these splits.

In Tab 1, we summarize *MOTSynth* statistics in relation to other real and synthetic datasets. In terms of size, the number of instances and labels, *MOTSynth* is superior to all the previously proposed datasets. For a detailed comparison, we refer to the supplementary material.

In contrast to VIPER [43] and GTA [44], *MOTSynth* focuses on crowded pedestrian scenarios. It is larger than pedestrian-focused JTA [24] and additionally provides instance segmentation and scene depth information. The key difference between JTA and *MOTSynth* lies in the volume of data, diversity of scenarios, and people variability that, as we experimentally show, allows us to bridge the synth-to-real gap.

MOTSynth contains 40M bounding boxes with tracking and segmentation mask labels, one to three orders of magnitude more compared to manually labeled MOTChallenge dataset suite (containing $292,733$ bounding boxes in MOT17, $1,652,040$ bounding boxes in MOT20, and $26,894$ segmentation masks in MOTS20 dataset). This difference is most prominent in the case of MOTS20, where pixel-precise labels for pedestrians are hard to obtain, even with semi-automated tools [75].

4. Experimental Evaluation

In this section, we experimentally validate whether *MOTSynth* can be used as a full proxy for (i) pedestrian detection (Sec. 4.2), (ii) pedestrian re-identification (ReID) (Sec. 4.3), (iii) multi-object tracking (Sec. 4.4), and (iv) multi-object tracking and segmentation (Sec. 4.6).

4.1. Experimental Setting

We evaluate all trained models on the MOTChallenge evaluation suite. To evaluate pedestrian detection and tracking, we use MOT17 [18] and MOT20 [19] datasets. We evaluate our ReID models on MOT17. Finally, we evaluate multi-object tracking and segmentation using MOTS20 dataset [75].

To understand how well the performance of models trained using synthetic data transfers to the real scenes of MOTChallenge, we train the models using the following datasets for comparison. We make a controlled study using the large-scale COCO dataset [49] for detection and tracking, and CrowdHuman [66] for tracking. For ReID, we employed two real-world ReID datasets Market1501 [83] and CUHK03 [47].

We further compare training on *MOTSynth* with other synthetic datasets depicting humans, namely, JTA [24] and VIPER [60]. To perform a fine-grained evaluation of *MOTSynth*-to-MOTChallenge transfer capabilities, we split *MOTSynth* into four (inclusive) subsets of 72, 144, 288 and 576 sequences, named *MOTSynth*-1 to *MOTSynth*-4. This also allows us to study the effect of the amount of data necessary to bridge the synth-to-real gap. For all experiments reported in this paper, we initialize the networks with ImageNet [20] pre-trained weights.

4.2. People Detection

To understand how training on *MOTSynth* compares to large-scale real-world datasets, we perform a series of experiments involving four heterogeneous object detectors: Faster RCNN [59] and Mask RCNN [34] as two-stage detectors, YOLOv3 [58] and CenterNet [85] as single-stage detectors. For each detector, we compare *MOTSynth* training against COCO training by testing on MOTChallenge.

We report the results in terms of average precision (AP), multi-object detection accuracy (MODA [7]), and the false-positive ratio measured by the number of false alarms per frame (FAF). In addition, we report precision, recall, and the absolute number of true positives (TP), false positives (FP), and false negatives (FN). For implementation details on these experiments, we refer to the supplementary. We will focus the discussion on AP as this is the most widely used detection metric.

Synth-to-real transfer. As can be seen in Tab. 2, by training the models on *MOTSynth*, we consistently outperform models trained on COCO. When evaluating these models on MOT17, we observe $+2.49$AP improvement for YOLOv3 with *MOTSynth*–4 compared to COCO, $+3.48$AP with CenterNet, $+2.3$AP with Faster R-CNN, and $+1.87$AP with Mask R-CNN. We conclude that the *improvements are consistent across different object detectors*.

These differences are further accentuated on MOT20,

		Dataset	AP ↑	MODA ↑	FAF ↓	TP ↑	FP ↓	FN ↓	Rec. ↑	Pr. ↑
MOT17	YOLOv3	COCO	69.76	62.02	1.25	47824	6650	18569	72.03	87.79
		MOTSynth–1	62.66	52.36	1.43	42378	7613	24015	63.83	84.77
		MOTSynth–2	63.08	56.67	1.22	44116	6489	22277	66.45	87.18
		MOTSynth–3	63.13	60.60	1.13	46264	6029	20129	69.68	88.47
		MOTSynth–4	**71.90**	**64.51**	**1.07**	**48500**	**5673**	**17893**	**73.05**	**89.53**
	CenterNet	COCO	67.01	44.38	3.37	47398	17935	18995	71.39	72.55
		MOTSynth–1	61.82	49.34	2.04	43626	10866	22767	65.71	80.06
		MOTSynth–2	62.32	54.90	**1.66**	45269	**8820**	21124	68.18	**83.69**
		MOTSynth–3	62.45	55.82	1.72	46177	9117	20216	69.55	83.51
		MOTSynth–4	**70.68**	**57.39**	1.81	**47748**	9646	**18645**	**71.92**	83.19
	Faster R-CNN	COCO	76.68	53.86	**3.45**	54127	**18364**	12266	81.52	74.67
		MOTSynth–1	76.80	39.02	5.19	53507	27603	12886	80.59	65.97
		MOTSynth–2	77.47	50.62	3.82	53893	20287	12500	81.17	72.65
		MOTSynth–3	78.31	49.75	4.22	**55474**	22441	**10919**	**83.55**	71.20
		MOTSynth–4	**78.98**	**54.96**	3.51	55121	18634	11272	83.02	**74.74**
	Mask R-CNN	COCO	76.96	55.55	3.31	54502	17620	11891	82.09	75.57
		MOTSynth–1	77.58	38.43	5.51	54817	29299	11576	82.56	65.17
		MOTSynth–2	77.88	50.01	4.09	54930	21724	11463	82.73	71.66
		MOTSynth–3	78.08	49.85	4.14	**55096**	21998	**11297**	**82.98**	71.47
		MOTSynth–4	**78.83**	**56.61**	**3.17**	54461	**16874**	11932	82.03	**76.35**

		Dataset	AP ↑	MODA ↑	FAF ↓	TP ↑	FP ↓	FN ↓	Rec. ↑	Pr. ↑
MOT20	YOLOv3	COCO	42.42	35.10	6.88	381635	61446	530602	41.84	86.13
		MOTSynth–1	35.83	31.15	**2.57**	307127	**22956**	605110	33.67	93.05
		MOTSynth–2	44.49	38.01	3.25	375739	29033	536498	41.19	92.83
		MOTSynth–3	44.68	42.89	3.56	423029	31797	489208	46.37	93.01
		MOTSynth–4	**53.69**	**48.87**	2.87	**471395**	25621	**440842**	**51.67**	**94.85**
	CenterNet	COCO	39.39	28.75	12.38	372835	110537	539402	40.87	77.13
		MOTSynth–1	43.35	30.84	16.21	426095	144781	486142	46.71	74.64
		MOTSynth–2	43.76	40.23	7.27	431932	64971	480305	47.35	**86.92**
		MOTSynth–3	34.08	24.29	**6.72**	281596	**60002**	630641	30.87	82.43
		MOTSynth–4	**51.70**	**42.18**	9.72	**471592**	86787	**440645**	**51.70**	84.46
	Faster R-CNN	COCO	43.67	40.55	5.90	422649	52698	489588	46.33	88.91
		MOTSynth–1	52.96	46.72	8.80	504790	78575	407447	55.34	86.53
		MOTSynth–2	52.56	46.96	7.91	498967	70609	413270	54.70	87.60
		MOTSynth–3	53.37	51.38	6.36	525547	56799	386690	57.61	90.25
		MOTSynth–4	**53.90**	**56.03**	**3.724**	**544416**	**33259**	**367821**	**59.67**	**94.25**
	Mask R-CNN	COCO	43.73	41.99	6.39	440081	57046	472156	48.24	88.52
		MOTSynth–1	52.75	44.98	10.28	502154	91819	410483	55.05	84.54
		MOTSynth–2	53.13	50.17	6.63	516896	59225	395341	56.66	89.72
		MOTSynth–3	53.51	52.27	5.76	528230	51408	384007	57.90	91.13
		MOTSynth–4	**54.03**	**55.69**	**4.11**	**544703**	**36715**	**367534**	**59.71**	**93.69**

Table 2: To perform synth-to-real control experiment, we train several object detector models on COCO dataset and on four *MOTSynth* subsets. We evaluate all models on MOTChallenge MOT17 (*left*) and MOT20 (*right*) detection datasets. We observe a clear trend with all object detectors: by purely training on synthetic data, we obtain better performance compared to training on a real-world dataset.

	Dataset	AP ↑	MODA ↑	FAF ↓	TP ↑	FP ↓	FN ↓	Rec. ↑	Pr. ↑
YOLOv3	VIPER	26.65	22.02	**0.16**	15447	**838**	50910	23.28	**94.85**
	JTA	53.18	48.77	0.79	36578	4200	29815	55.09	89.70
	MOTSynth–256	62.99	62.31	0.58	44458	3090	21935	66.96	93.50
	***MOTSynth*-full**	71.90	64.51	1.07	**48500**	5673	**17893**	**73.05**	89.53
CenterNet	VIPER	44.58	36.92	**1.24**	31122	**6611**	35271	46.88	**82.48**
	JTA	60.15	45.38	2.32	42435	12308	23958	63.91	77.52
	MOTSynth–256	61.82	50.11	2.03	44067	10795	22326	66.37	80.32
	***MOTSynth*-full**	**70.49**	**55.25**	2.11	**47883**	11204	**18510**	**72.12**	81.04
Faster R-CNN	VIPER	60.93	42.87	**2.87**	43707	**15241**	**10593**	65.82	74.14
	JTA	69.69	38.38	5.12	52726	27242	13667	65.93	**79.41**
	MOTSynth–256	78.61	**58.65**	3.10	**55441**	16504	10952	**83.50**	77.06
	***MOTSynth*-full**	**78.98**	54.96	3.51	55121	18634	11272	83.02	74.74

Table 3: Comparison on MOT17 against synthetic datasets.

where we observe a consistent and remarkable improvement of +10.97AP and +12.31AP on YOLOv3 and CenterNet, respectively, and +10.23AP (Faster R-CNN) and +10.3AP (Mask R-CNN).

We observe that for both MOT17 and MOT20, single-stage detectors benefit from the full *MOTSynth* dataset, while two-stage detectors improve marginally from *MOTSynth*–1 to *MOTSynth*–4 (+0.12 and +0.62 improvement on MOT17 and +0.94 and +1.28 improvement on MOT20 in terms of AP with Faster R-CNN and Mask R-CNN, respectively). A possible explanation is that single-stage detectors have to learn a more complex function than two-stage detectors, splitting the problem into two simpler tasks and consequently requiring more data to train effectively.

Data volume vs. diversity. To understand the impact of increasing dataset diversity versus increasing the amount of training data, we perform the following experiment. We keep the number of training images fixed and sample images from sequences using two different sampling rates ($1/60$ and $1/10$). The higher the sampling rate, the more images we sample from a given sequence, and vice versa. Thus, by decreasing the sampling rates, we increase the diversity as we sample images from a larger number of different sequences. When evaluating on the smallest *MOTSynth*–1 subset, we observe a clear trend: *diversity matters*. When sampling with $1/10$ rate, we reach 76.8AP and match the COCO model's performance (76.69AP). However, with denser and therefore less diverse sampling, this is not the case (70AP). We report detailed results for different object detectors in the supplementary.

Comparison of synthetic datasets. As demonstrated, we were able to bridge the synth-to-real gap using *MOTSynth*. *Is this also the case for other synthetic datasets?* To answer this question, we conduct a similar experiment by training models on VIPER [43] and JTA [24] datasets. As can be seen in Tab. 3, *MOTSynth*-based training clearly outperforms alternative synthetic datasets consistently. In particular, YOLOv3 trained on ***MOTSynth*-full** outperforms VIPER-trained models by +45.25AP and JTA trained models by +18.72AP. We observe a similar trend with CenterNet. We obtain +25.91AP improvement with ***MOTSynth*-full**-trained models over the VIPER model and +10.34AP improvement over the JTA model. Finally, with Faster R-CNN, we observe +18.05 improvement over the VIPER model and +18.29 over the JTA model.

These observations beg the following question: *What is the advantage of MOTSynth over pedestrian-oriented JTA* – is it the diversity or sheer amount of data? To answer this question, we conduct the following experiment. We train each detector using the subset of *MOTSynth*, *MOTSynth*–256 (*i.e.*, *MOTSynth*–1), containing only 256 sequences, generated from the 128 screenplays provided by the authors of [24]. The only difference between JTA and *MOTSynth*–256 is in people appearance variation – high per-

	Dataset	AP ↑	MODA ↑	FAF ↓	TP ↑	FP ↓	FN ↓	Rec. ↑	Pr. ↑
MOT17	ZIZOM [48]	**0.81**	72.0	2.2	95414	12990	19139	83.3	88.0
	SDP [79]	**0.81**	**76.9**	**1.3**	95699	**7599**	18865	83.5	**92.6**
	DPM [26]	0.61	31.2	7.1	78007	42308	36557	68.1	64.8
	FRCNN [59]	0.72	68.5	1.7	88601	10081	25963	77.3	89.8
	FRCNN *MOTSynth*	0.80	66.7	3.7	98164	21748	16400	81.9	83.7
	FRCNN *MOTSynth* + FT	0.80	71.0	3.5	**102341**	20989	**12223**	**89.3**	83.0
MOT20	GNN_SDT [76]	**0.81**	**79.3**	7.1	**304236**	31677	**39288**	**88.6**	90.6
	VIPeD20 [14]	0.80	46.0	31.1	297101	139111	46277	86.5	68.1
	FRCNN *MOTSynth*	0.62	52.0	6.3	206902	28202	136622	60.2	88.0
	FRCNN *MOTSynth* + FT	0.72	63.3	**5.2**	241056	**23465**	102468	70.2	**91.1**

Table 4: We train Faster R-CNN on *MOTSynth* with and without fine-tuning and evaluate on MOTChallenge MOT17 and MOT20 pedestrian detection test sets.

son appearance variety was one of the key goals when generating *MOTSynth* sequences. As can be seen, with YOLOv3 and Faster R-CNN *MOTSynth*–256 models, we obtain $+9.81AP$ and $+8.92AP$ improvements over JTA trained models. *This confirms that the MOTSynth diversity in terms of people appearance is a crucial ingredient for bridging the domain gap.*

Benchmark results. Finally, we evaluate our *MOTSynth*-trained detection models' generalization capability by submitting our results to the MOTChallenge MOT17&MOT20 benchmarks. We evaluate two variants: no fine-tuning, *i.e.*, trained only on *MOTSynth*, and with fine-tuning (+ FT) on the respective MOTChallenge dataset. We summarize our results in Tab. 4. As can be seen, on MOT17, we outperform (FRCNN *MOTSynth*, 0.8AP) the baseline Faster R-CNN (FRCNN, 0.72AP) by $+0.08$AP. Interestingly, fine-tuning on the MOTChallenge training set does not significantly impact the *MOTSynth* model in terms of AP. It does, however, improve in terms of MODA (66.7 vs. 71 MODA after fine-tuning), for which a specific threshold needs to be selected. During the experiments, we kept the original threshold. It is important to note that more recent object detectors, ZIZOM [48] and SDP [79] only marginally improve over our *MOTSynth*-trained Faster R-CNN models ($+0.01AP$).

Unlike MOT17, fine-tuning has a significant effect on MOT20 ($+0.1AP$): we assume this is because in *MOTSynth* we do not have the extremely crowded scenes that are the focus of MOT20. Generating denser synthetic sequences could further help to bridge the gap on MOT20 and remains our future work. Finally, we note that detectors specialized in pedestrian detection in crowded scenes [14, 76] outperform our fine-tuned *MOTSynth* Faster R-CNN model by only $+0.08AP$.

4.3. Person Re-Identification

To evaluate the re-identification (ReID) model performance, we train three models, (i) trained on Market1501 [83], (ii) trained on Market1501 [83] and CUHK03 [47] and, finally, trained *only* on four subsets *MOTSynth*. We evaluate all three models out-of-the-box (without fine-tuning) on the MOTChallenge MOT17 dataset by treating each sequence as a separate dataset. To do so, we randomly select one ground truth box per track to obtain a query set and use the remaining set of boxes, sampled at 10 FPS, as a gallery set. We compute standard ReID metrics for every sequence: mean average precision (mAP) and Rank-1 accuracy, and report their average overall sequences. All models are trained with a ResNet-50 backbone, followed by a fully connected layer and a standard cross-entropy loss. For implementation details, we refer to the supplementary.

	Dataset	Split	mAP	Rank1
Real	Market1501 [83]	–	64.6	91.9
	Market1501 [83] + CUHK03 [47]	–	69.1	91.9
Synthetic	*MOTSynth*	1	71.3	91.4
		2	73.1	91.8
		3	74.2	92.6
		4	**75.2**	**92.8**

Table 5: Person ReID experiments on MOT17.

As can be seen in Tab. 5, by training purely on *MOTSynth* data using the first split, we already outperform models trained on real data in terms of mAP ($+6.9$ for Market1501 and $+2.5$ for combined datasets). In terms of Rank1, we obtain $+1.6$ relative to the Market1501-only model. However, when training on *MOTSynth* using the first two splits (50% of total data), we notice an improvement of $+8.6$ and $+4.2$ in terms of mAP and $+3.5$ and $+1$ in terms of Rank1, respectively. *This suggests synthetic datasets can be used as a full replacement for ReID datasets, which are often a subject of controversy [33].*

4.4. Multi-object Tracking

In this section, we analyze the value of *MOTSynth* for the task of pedestrian multi-object tracking. We report CLEAR-MOT [7] and IDF1 [62] metrics and focus the analysis on the most widely used Multiple Object Tracking Accuracy (MOTA) and Identity F1 Score (IDF1). We experiment with two different trackers, Tracktor [6] and recently proposed CenterTrack [84]. We evaluate all our models on the most widely used pedestrian tracking dataset, MOTChallenge MOT17 [18], with and without fine-tuning (**FT**) on the MOT17 training set. Following the CenterTrack validation scheme, we fine-tune the networks using only the first half of MOT17 sequences and validate on the second half.

Tracktor. We train detection/tracking [6] model on (i) COCO dataset [49] and (ii) full *MOTSynth* dataset. We note that for Tracktor, we do not need to do any training on sequences, as this method leverages bounding box regression functionality to follow targets. Tracktor also relies on the ReID models to bridge trajectory gaps. To this end, we experiment with two ReID models, one trained on real data (Market1501) and one trained on synthetic data (*MOTSynth*).

	Dataset	FT	ReID	MOTA ↑	MOTP ↑	IDF1 ↑	TP ↑	FP ↓	FN ↓	IDS ↓
Tracktor [6]	COCO	✗	✗	43.5	0.192	49.6	26783	**2816**	27259	467
	COCO	✗	Market1501	44.0	0.192	55.1	26783	**2816**	27259	179
	COCO	✓	Market1501	48.3	0.193	58.1	29218	27259	24824	185
	MOTSynth	✗	✗	45.0	0.197	51.2	28749	3992	25293	458
	MOTSynth	✗	Market1501	45.5	0.197	56.8	28749	3992	25293	161
	MOTSynth	✗	*MOTSynth*	45.5	0.197	56.8	28749	3992	25293	**160**
	MOTSynth	✓	✗	49.8	**0.199**	53.8	**30588**	3264	**23454**	411
	MOTSynth	✓	Market1501	**50.3**	**0.199**	59.8	**30588**	3264	**23454**	167
	MOTSynth	✓	*MOTSynth*	**50.3**	**0.199**	**59.9**	**30588**	3264	**23454**	165
CenterTr. [84]	ImageNet	✓	–	60.7	0.190	62.7	35443	2179	18447	564
	CrowdHuman	✗	–	52.2	**0.218**	53.8	**32486**	3604	21404	728
	CrowdHuman	✓	–	66.1	0.179	64.2	38604	2442	15286	528
	MOTSynth	✗	–	54.3	0.205	57.7	33504	3601	20386	666
	MOTSynth	✓	–	**67.9**	0.179	**66.5**	38681	**1606**	**15209**	**508**

Table 6: Multi-object tracking results performed on MOT17 training set.

CenterTrack. For CenterTrack [84] we report the results from the paper. In particular, we report (i) CenterTrack model trained on MOT17 directly, the model trained on (ii) the CrowdHuman dataset [66] using a static-image training scheme, and finally, we (iii) report the results we obtain with CenterTrack trained on *MOTSynth* instead of CrowdHuman. We train only on the *MOTSynth*–1 subset using full sequences (every frame). In this case, we train for four epochs using the same train/eval hyperparameters from the CenterTrack paper.

Fine-tuning. We also evaluate how fine-tuning on MOTChallenge affects the final performance of each model. In the case of CenterTrack, we employ a slightly modified pre-training scheme to fully utilize the scene diversity of *MOTSynth*. Instead of training only on *MOTSynth*–1 subset, we train on all *MOTSynth* sequences. However, due to computational constraints, we only use a subset of frames ($1/8$ of each video) within each sequence. This way, we increase the scene diversity while keeping the training time reasonable. After training with this subset of *MOTSynth* for 10 epochs, we fine-tune our network on MOT17 sequences for 28 epochs. Throughout fine-tuning and validation, we use the same training and evaluation hyper-parameters as reported in [84, 6]. While [84] reports performing 70 epochs on the CrowdHuman dataset, we stopped training on *MOTSynth* earlier as we observed our model started over-fitting. For further implementation details of these experiments, we refer to the supplementary.

Results. We report our findings in Tab. 6. First, we analyze Tracktor performance. When **not performing any fine-tuning or using ReID model**, we obtain 45.0 MOTA and 51.2 IDF1 with our *MOTSynth* trained model, yielding $+3.5$ MOTA and $+1.6$ IDF1 improvement over the COCO-trained model (43.5 MOTA and 49.6 IDF1). After we **fine-tune** both models on MOT17, the *MOTSynth*-trained model (49.8 MOTA, 53.8 IDF1) improves by $+4.8$ in terms of MOTA and $+2.6$ in terms of IDF1. Similarly, the fine-tuned COCO trained model (48.3 MOTA, 58.1 IDF1) improves by $+4.8$ MOTA and $+8.5$ IDF1. After fine-tuning, the improvement of *MOTSynth* over COCO increases by $+1.5$ MOTA and $+4.3$ IDF1, suggesting that *MOTSynth is more suitable for pre-training pedestrian detection and tracking models compared to COCO dataset.*

	Dataset	sMOTSA ↑	MOTSA ↑	MOTSP ↑	IDF1 ↑	TP ↑	FP ↓	FN ↓	IDS ↓
[75]	Several	52.74	66.90	80.16	51.19	19202	894	7692	315
[6]	COCO	55.58	68.80	**81.93**	63.24	19677	1016	7217	159
	MOTSynth	55.54	68.73	**81.93**	63.09	19626	987	7268	155
	MOTSynth †	**56.10**	**69.52**	81.67	**67.53**	**19690**	**850**	**7204**	**143**
[35]	COCO	53.59	67.53	**81.52**	73.31	20279	2026	6615	92
	MOTSynth	**54.09**	**68.07**	81.49	**73.48**	**20304**	**1912**	**6590**	**86**
[84]	COCO	**53.93**	**67.49**	81.68	**58.4**	**19919**	**1488**	**6975**	279
	MOTSynth	53.88	67.37	**81.74**	58.19	19876	1497	7018	**260**
[10]	COCO	48.50	62.41	81.34	69.03	20061	3177	6833	99
	MOTSynth	**49.17**	**63.03**	**81.44**	**69.40**	**20079**	**3047**	**6815**	**82**

Table 7: Multi-object tracking and segmentation. Masks were generated using Mask R-CNN model, trained on COCO and *MOTSynth*. Baselines: Track R-CNN [75], Tracktor [6], Lift_T [35], CenterNet [84], MPNTrack [10]

When **using ReID models**, we observe consistent improvements on both *MOTSynth* and COCO models. In particular, we observe a consistent improvement in terms of MOTA, which we attribute to a significant reduction ($\sim$250) in the number of IDS. Interestingly, we observe identical improvements with both ReID models, trained on *MOTSynth* and Market1501, and *conclude that the ReID model, trained on MOTSynth is an adequate replacement for models trained on the real data.*

CenterTrack reports 60.7 MOTA and 62.7 IDF1 when training directly on MOT17, and 52.2 MOTA and 53.8 IDF1 on MOT17 when training on CrowdHuman using the static-image training scheme and evaluating directly on MOT17. We obtain notably better results when training on *MOTSynth* and evaluating on MOT17: 54.3 MOTA ($+2.1$) and 57.7 IDF1 ($+3.9$). After fine-tuning the CrowdHuman model on MOT17, we obtain 66.1 MOTA and 64.2 IDF1, and even better performance when fine-tuning *MOTSynth* model: 67.9 MOTA and 66.5 IDF1.

Overall, synthetic training always performs favorably, indicating that *MOTSynth* can completely replace manually annotated datasets while increasing performance.

4.5. Multi-object Tracking and Segmentation

In this section, we analyze multi-object tracking and segmentation (MOTS) [75]. We report CLEARMOT metrics [7], adapted for MOTS as proposed in [75]. Different from MOT, object tracks are localized with segmentation masks instead of bounding boxes.

For these experiments, we first take the tracking outputs of several state-of-the-art MOT methods that use public SDP [79] detector and predict segmentation masks with a Mask R-CNN, trained either on COCO or *MOTSynth*. We perform the experiments on the MOTS20 train set and perform no fine-tuning on these sequences. In particular, we analyze Tracktor [6], Lift_T [35], CenterNet [84], and MP-

NTrack [10]. For Tracktor, we perform an additional experiment. We turn Mask R-CNN trained on *MOTSynth* into a Tracktor that directly produces pixel-precise tracking output (denoted with †). For reference, we also report TrackR-CNN [75], trained on COCO [49], Mapillary Vistas [55] and MOTS20.

We report our findings in Tab. 7. Comparing COCO and *MOTSynth* models, we observe 0.5 point increase in sMOTSA, MOTSA, and IDF1 for Lif_T and MPNTrack in favor of *MOTSynth*. However, for CenterTrack and Tracktor, we observe a minimal drop in performance (-0.1 points). Interestingly, Mask R-CNN Tracktor (†), trained directly on *MOTSynth* outperforms Tracktor_v2++ for $+4$ IDF1, $+0.5$ sMOTSA and $+0.7$ MOTSA. This is our best-performing entry on MOTS20. It is important to note that this model is trained **only** on synthetic data, whereas other methods reported were trained using MOTChallenge, or several datasets in the case of TrackR-CNN. We report implementation details for these experiments in the supplementary.

4.6. Benchmark Results

Finally, we evaluate our models on MOTChallenge MOT17, MOT20, and MOTS20 test sets using the public benchmark.

MOT17. As shown in Tab. 8, we obtain highly competitive results when solely training using synthetic data. In fact, on MOT17, Tracktor–*MOTSynth* outperforms Tracktor, trained on COCO, and fine-tuned on MOT17 by $+3.4$ MOTA and $+4.6$ IDF1! Fine-tuning on MOT17 further improves metrics by $+2.2$ MOTA and $+1.9$ IDF1. Similarly, CenterTrack trained only on synthetic data achieves competitive results (59.7 MOTA, 52.0 IDF1). The model, fine-tuned on MOT17, further improves performance, establishing a new state of the art with 65.1 MOTA and 57.9 IDF1.

MOT20. On MOT20, Tracktor and CenterTrack trained solely on *MOTSynth* are not yet on-par with state-of-the-art. However, when fine-tuned on MOT20, these models surpass Tracktorv2 by $+3.9$ MOTA and $+0.1$ IDF1.

MOTS20. We show MOTS20 benchmark results[3] in Tab. 9. Our Tracktor† Mask R-CNN trained **only** on synthetic data significantly outperforms TrackR-CNN [75], that is trained on COCO, Mapillary Vistas [55] and MOTS20 training set. In particular, we improve sMOTSA for $+14.45$, MOTSA for $+15.04$, MOTSP for $+3,47$ and IDF1 for $+21.47$. This confirms our intuition that *MOTSynth* is especially beneficial in the scarce data regime, as is the case for the MOTS task.

These experiments confirm that top-ranked MOTChallenge models can be trained purely on synthetic data on the MOT17 and MOTS datasets to achieve state-of-the-art results. However, on the MOT20 dataset, fine-tuning is still needed to reach state-of-the-art results. We assume that this is due to the fact that synthetic sequences more closely resemble MOT17 sequences than extremely crowded MOT20 sequences. This hints that *MOTSynth* has future potential in closing this gap by simulating similarly dense environments.

[3]In the time of submission, there is only one published entry in MOTS20 benchmark.

	Method	MOTA ↑	MOTP ↑	IDF1 ↑	FP ↓	FN ↓	IDS ↓
MOT17	Tracktor-*MOTSynth*	56.9	78.0	56.9	20852	220273	2012
	Tracktor-*MOTSynth* + FT	59.1	79.3	58.8	22231	206062	2323
	Tracktor [6]	53.5	78.0	52.3	12201	248047	2072
	Tracktorv2 [6]	56.3	78.8	55.1	**8866**	235449	1987
	CenterTrack-*MOTSynth*	59.7	77.4	52.0	39707	181471	6035
	CenterTrack-*MOTSynth* + FT	**65.1**	**79.9**	57.9	11521	**180901**	4377
	CenterTrack [84]	61.5	78.9	59.6	14076	200672	2583
	Lif_T [35]	60.5	78.3	65.6	14966	206619	1189
	LPC [16]	59.0	78.0	**66.8**	23102	206948	**1122**
	MPNTrack [10]	58.8	78.6	61.7	17413	213594	1185
MOT20	Tracktor-*MOTSynth*	43.7	75.1	39.7	15933	271814	3467
	Tracktor-*MOTSynth* + FT	56.5	78.8	52.8	11377	211772	1995
	Tracktorv2 [6]	52.6	79.9	52.7	**6930**	236680	1648
	CenterTrack-*MOTSynth*	39.7	72.9	37.2	47066	259274	5872
	CenterTrack-*MOTSynth* + FT	41.9	**80.2**	38.2	36594	258874	5313
	MPNTrack [10]	**57.6**	79.0	59.1	16953	**201384**	**1210**
	LPC [16]	56.3	79.7	**62.5**	11726	213056	1562
	SORT20 [8]	42.7	78.5	45.1	27521	264694	4470

Table 8: Benchmark results on MOT17 and MOT20. We refer to supplementary for the full version.

Method	sMOTSA ↑	MOTSA ↑	MOTSP ↑	IDF1 ↑	TP ↑	FP ↓	FN ↓	IDS ↓
Tracktor†-*MOTSynth*	**55.1**	**70.2**	**79.6**	**63.9**	**23994**	**1128**	**8275**	**200**
TrackRCNN [75]	40.6	55.2	76.1	42.4	19628	1261	12641	567

Table 9: Benchmark results on MOTS20 dataset.

5. Conclusion

We presented *MOTSynth*, a massive synthetic dataset for pedestrian detection and tracking in urban scenarios. We experimentally demonstrated that synthetic data could completely substitute real data for high-level in-the-wild scenarios, such as pedestrian detection, re-identification tracking, and segmentation. Remarkably, we obtained state-of-the-art results on the MOTChallenge MOT17 dataset by training recent methods using solely synthetic data. We believe this paper will pave the road for future efforts in replacing costly data collection with synth in other domains.

Acknowledgments. The authors would like to thank Tim Meinhardt for his helpful comments on our manuscript. For partial funding of this project, GB, AO, OC, and LLT would like to acknowledge the Humboldt Foundation through the Sofja Kovalevskaja Award. MF, GM, RG, SC, and RC would like to acknowledge the InSecTT project, funded by the ECSEL Joint Undertaking (JU) under GA 876038. The JU receives support from the EU H2020 Research and Innovation programme and AU, SWE, SPA, IT, FR, POR, IRE, FIN, SLO, PO, NED, TUR. The document reflects only the author's view and the Commission is not responsible for any use that may be made of the information it contains.

References

[1] Script hook v. http://www.dev-c.com/gtav/scripthookv/. Accessed: 2021-03-08. 3

[2] 2018 reform of eu data protection rules. https://gdpr-info.eu, 2018. 1

[3] Giuseppe Amato, Luca Ciampi, Fabrizio Falchi, Claudio Gennaro, and Nicola Messina. Learning pedestrian detection from virtual worlds. In *Int. Conf. on Image Analysis and Process.*, 2019. 1, 2, 3

[4] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 4

[5] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *Int. J. Comput. Vis.*, 2011. 2

[6] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixé. Tracking without bells and whistles. In *Int. Conf. Comput. Vis.*, 2019. 1, 2, 6, 7, 8

[7] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008. 4, 6, 7

[8] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *IEEE Int. Conf. Image Process.*, 2016. 8

[9] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 1, 2

[10] Guillem Brasó and Laura Leal-Taixé. Learning a neural solver for multiple object tracking. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1, 2, 7, 8

[11] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *Eur. Conf. Comput. Vis.*, 2012. 2

[12] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

[13] Qi Chu, Wanli Ouyang, Hongsheng Li, Xiaogang Wang, Bin Liu, and Nenghai Yu. Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism. *Int. Conf. Comput. Vis.*, 2017. 1

[14] Luca Ciampi, Nicola Messina, Fabrizio Falchi, Claudio Gennaro, and Giuseppe Amato. Virtual to real adaptation of pedestrian detectors. *Sensors*, 20(18):5250, 2020. 6

[15] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 2

[16] Peng Dai, Renliang Weng, Wongun Choi, Changshui Zhang, Zhangping He, and Wei Ding. Learning a proposal classifier for multiple object tracking. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 8

[17] Achal Dave, Tarasha Khurana, Pavel Tokmakov, Cordelia Schmid, and Deva Ramanan. TAO: A large-scale benchmark for tracking any object. In *Eur. Conf. Comput. Vis.*, 2020. 2

[18] Patrick Dendorfer, Aljoša Ošep, Anton Milan, Konrad Schindler, Daniel Cremers, Ian Reid, and Stefan Roth Laura Leal-Taixé. Motchallenge: A benchmark for single-camera multiple target tracking. *Int. J. Comput. Vis.*, 2020. 2, 4, 6

[19] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv preprint arXiv:2003.09003*, 2020. 2, 4

[20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009. 4

[21] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015. 1, 3

[22] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vis.*, 2015. 2

[23] Matteo Fabbri, Fabio Lanzi, Simone Calderara, Stefano Alletto, and Rita Cucchiara. Compressed volumetric heatmaps for multi-person 3d pose estimation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 3

[24] Matteo Fabbri, Fabio Lanzi, Simone Calderara, Andrea Palazzi, Roberto Vezzani, and Rita Cucchiara. Learning to detect and track visible and occluded body joints in a virtual world. In *Eur. Conf. Comput. Vis.*, 2018. 2, 3, 4, 5

[25] Kuan Fang, Yu Xiang, Xiaocheng Li, and Silvio Savarese. Recurrent autoregressive networks for online multi-object tracking. In *IEEE Wint. Conf. App. Comput. Vis.*, 2018. 1

[26] P. Felzenszwalb, D. Mcallester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2008. 6

[27] Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 3

[28] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2012. 2

[29] Thomas Golda, Tobias Kalb, Arne Schumann, and Jürgen Beyerer. Human pose estimation for real-world crowded scenarios. In *IEEE Int. Conf. Advan. Video Signal Based Surv.*, 2019. 3

[30] Ankur Handa, Richard A Newcombe, Adrien Angeli, and Andrew J Davison. Real-time camera tracking: When is high frame-rate best? In *Eur. Conf. Comput. Vis.*, 2012. 2

[31] Ankur Handa, Viorica Patraucean, Vijay Badrinarayanan, Simon Stent, and Roberto Cipolla. Understanding real world

indoor scenes with synthetic data. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 3

[32] Ankur Handa, Thomas Whelan, John McDonald, and Andrew J Davison. A benchmark for rgb-d visual odometry, 3d reconstruction and slam. In *IEEE Int. Conf. Robotics and Autom.*, 2014. 2

[33] Adam Harvey and Jules LaPlace. Megapixels: origins, ethics, and privacy implications of publicly available face recognition image datasets. *Megapixels*, 2019. 1, 2, 6

[34] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *Int. Conf. Comput. Vis.*, 2017. 4

[35] Andrea Hornakova, Roberto Henschel, Bodo Rosenhahn, and Paul Swoboda. Lifted disjoint paths with application in multiple object tracking. In *IEEE Int. Conf. Mach. Learn.*, 2020. 7, 8

[36] Hou-Ning Hu, Qi-Zhi Cai, Dequan Wang, Ji Lin, Min Sun, Philipp Krähenbühl, Trevor Darrell, and Fisher Yu. Joint monocular 3d vehicle detection and tracking. In *Int. Conf. Comput. Vis.*, 2019. 3

[37] Yuan-Ting Hu, Hong-Shuo Chen, Kexin Hui, Jia-Bin Huang, and Alexander G Schwing. Sail-vos: Semantic amodal instance level video object segmentation-a synthetic dataset and baselines. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 3

[38] Yanru Huang, Feiyu Zhu, Zheni Zeng, Xi Qiu, Yuan Shen, and Jianan Wu. Sqe: a self quality evaluation metric for parameters optimization in multi-object tracking. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1

[39] Matthew Johnson-Roberson, Charles Barto, Rounak Mehta, Sharath Nittur Sridhar, Karl Rosaen, and Ram Vasudevan. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? *IEEE Int. Conf. Robotics and Autom.*, 2017. 3

[40] Biliana Kaneva, Antonio Torralba, and William T Freeman. Evaluation of image features using a photorealistic virtual world. In *Int. Conf. Comput. Vis.*, 2011. 2

[41] Chanho Kim, Fuxin Li, Arridhana Ciptadi, and James M. Rehg. Multiple hypothesis tracking revisited. In *Int. Conf. Comput. Vis.*, 2015. 1

[42] Chanho Kim, Fuxin Li, and James M. Rehg. Multi-object tracking with neural gating using bilinear lstm. In *Eur. Conf. Comput. Vis.*, 2018. 1

[43] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Video panoptic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2, 3, 4, 5

[44] Philipp Krähenbühl. Free supervision from video games. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 1, 2, 3, 4

[45] Laura Leal-Taixé, Cristian Canton-Ferrer, and Konrad Schindler. Learning by tracking: Siamese cnn for robust target association. *CVPR Workshops*, 2016. 1

[46] Bastian Leibe, Konrad Schindler, Nico Cornelis, and Luc Van Gool. Coupled object detection and tracking from static cameras and moving vehicles. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(10):1683–1698, 2008. 3

[47] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2014. 2, 4, 6

[48] Chunze Lin, Jiwen Lu, Gang Wang, and Jie Zhou. Graininess-aware deep feature learning for pedestrian detection. In *Eur. Conf. Comput. Vis.*, 2018. 6

[49] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Eur. Conf. Comput. Vis.*, 2014. 2, 4, 6, 8

[50] Jonathon Luiten, Paul Voigtlaender, and Bastian Leibe. Premvos: Proposal-generation, refinement and merging for video object segmentation. In *Asian Conf. Comput. Vis.*, 2018. 3

[51] Javier Marin, David Vázquez, David Gerónimo, and Antonio M López. Learning appearance in virtual scenarios for pedestrian detection. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2010. 3

[52] Nikolaus Mayer, Eddy Ilg, Philipp Fischer, Caner Hazirbas, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. What makes good synthetic training data for learning disparity and optical flow estimation? *Int. J. Comput. Vis.*, 2018. 2

[53] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 2

[54] Anton Milan, Laura. Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. MOT16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016. 1, 2, 4

[55] Gerhard Neuhold, Tobias Ollmann, S Rota Bulo, and Peter Kontschieder. The Mapillary Vistas dataset for semantic understanding of street scenes. In *Int. Conf. Comput. Vis.*, 2017. 8

[56] Aljoša Ošep, Wolfgang Mehner, Markus Mathias, and Bastian Leibe. Combined image- and world-space tracking in traffic scenes. In *IEEE Int. Conf. Robotics and Autom.*, 2017. 3

[57] Abhishek Patil, Srikanth Malla, Haiming Gang, and Yi-Ting Chen. The h3d dataset for full-surround 3d multi-object detection and tracking in crowded urban scenes. In *IEEE Int. Conf. Robotics and Autom.*, 2019. 2

[58] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 4

[59] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Adv. Neural Inform. Process. Syst.*, 2015. 4, 6

[60] Stephan R Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *Int. Conf. Comput. Vis.*, 2017. 2, 3, 4

[61] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *Eur. Conf. Comput. Vis.*, 2016. 3

[62] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *Eur. Conf. Comput. Vis.*, 2016. 1, 6

[63] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 3

[64] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 2015. 2

[65] Samuel Schulter, Paul Vernaza, Wongun Choi, and Manmohan Krishna Chandraker. Deep network flow for multi-object tracking. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 1

[66] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018. 4, 7

[67] Toby Sharp, Cem Keskin, Duncan Robertson, Jonathan Taylor, Jamie Shotton, David Kim, Christoph Rhemann, Ido Leichter, Alon Vinnikov, Yichen Wei, et al. Accurate, robust, and flexible real-time hand tracking. In *ACM conf. on human factors in comput. sys.*, 2015. 3

[68] Jamie Shotton, Ross Girshick, Andrew Fitzgibbon, Toby Sharp, Mat Cook, Mark Finocchio, Richard Moore, Pushmeet Kohli, Antonio Criminisi, Alex Kipman, et al. Efficient human pose estimation from single depth images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2012. 3

[69] Jeany Son, Mooyeol Baek, Minsu Cho, and Bohyung Han. Multi-object tracking with quadruplet convolutional neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 1

[70] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

[71] ShiJie Sun, Naveed Akhtar, XiangYu Song, HuanSheng Song, Ajmal Mian, and Mubarak Shah. Simultaneous detection and tracking with motion modelling for multiple object tracking. In *Eur. Conf. Comput. Vis.*, 2020. 3

[72] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. 2017. 2, 3

[73] Jonathan Tremblay, Aayush Prakash, David Acuna, Mark Brophy, Varun Jampani, Cem Anil, Thang To, Eric Cameracci, Shaad Boochoon, and Stan Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *CVPR Workshops*, 2018. 2

[74] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 3

[75] Paul Voigtlaender, Michael Krause, Aljoša Ošep, Jonathon Luiten, B.B.G Sekar, Andreas Geiger, and Bastian Leibe. MOTS: Multi-object tracking and segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 1, 2, 4, 7, 8

[76] Yongxin Wang, Xinshuo Weng, and Kris Kitani. Joint detection and multi-object tracking with graph neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 6

[77] Longyin Wen, Dawei Du, Zhaowei Cai, Zhen Lei, Ming-Ching Chang, Honggang Qi, Jongwoo Lim, Ming-Hsuan Yang, and Siwei Lyu. Ua-detrac: A new benchmark and protocol for multi-object detection and tracking. *arXiv preprint arXiv:1511.04136*, 2015. 2

[78] Yihong Xu, Aljoša Ošep, Yutong Ban, Radu Horaud, Laura Leal-Taixé, and Xavier Alameda-Pineda. How to train your deep multi-object tracker. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1

[79] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 6, 7

[80] Junbo Yin, Wenguan Wang, Qinghao Meng, Ruigang Yang, and Jianbing Shen. A unified object motion and affinity model for online multi-object tracking. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1

[81] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

[82] Zichao Zhang, Henri Rebecq, Christian Forster, and Davide Scaramuzza. Benefit of large field-of-view cameras for visual odometry. In *IEEE Int. Conf. Robotics and Autom.*, 2016. 2

[83] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *Int. Conf. Comput. Vis.*, 2015. 2, 4, 6

[84] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *Eur. Conf. Comput. Vis.*, 2020. 6, 7, 8

[85] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 4